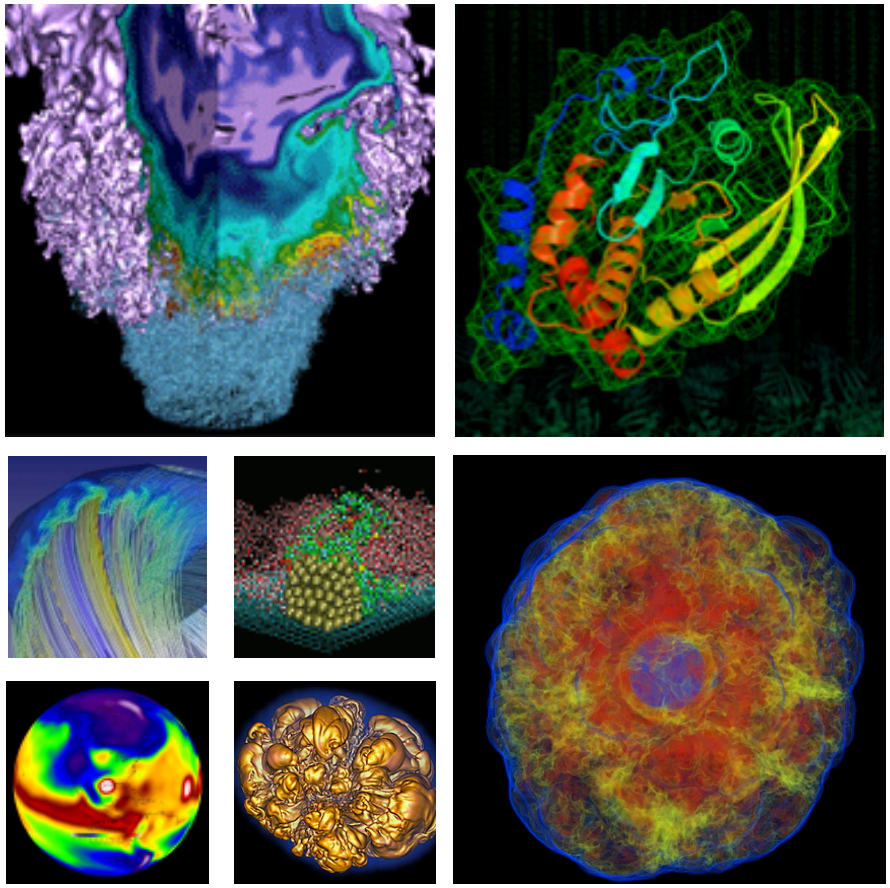
Computing Environment



Scott French
NERSC User Services Group

New User Training
August 13, 2015

NERSC 40 YEARS at the FOREFRONT 1974-2014

U.S. DEPARTMENT OF ENERGY | Office of Science

BERKELEY LAB
Lawrence Berkeley National Laboratory

Node Types

- **Login nodes**
  - Shared with other users
  - Code compilation, job preparation and submission
- **MOM nodes (Cray machines)**
  - Shared with other users
  - Where batch script executes
  - Will go away when we transition to SLURM
- **Compute nodes**
  - Not shared (except serial queues)

Login Node Configuration

- **Edison**
  - Twelve nodes
    - 16 cores, 2.0 GHz Intel Sandy Bridge, 512 GB
- **Hopper**
  - Eight nodes
    - 16 cores, 2.4 GHz AMD Opteron, 128GB
  - Four nodes
    - 32 cores, 2.0 GHz AMD Opteron, 128GB
- **Genepool**
  - Four nodes
    - 8 cores, 2.3 GHz Intel Sandy Bridge, 32 GB
- **PDSF**
  - Three nodes
    - 16 cores, 2.6 GHz Intel Sandy Bridge, 125 GB

Login Node Access

- **Connect (via ssh) to *load balancer***
  - `% ssh edison.nersc.gov`
  - `% ssh hopper.nersc.gov`
  - `% ssh genepool.nersc.gov`
  - `% ssh pdsf.nersc.gov`
- **Load balancer selects login node based on:**
  - Number of connections
  - Memory of previous connections from same IP

Login Node Usage

- **Login nodes are shared by many users, all the time**
- **Edit files, compile programs, submit batch jobs**
- ***Some* post-processing/data analysis**
  - IDL, MATLAB, NCL, python, etc.
- ***Some* file transfers**
  - Use data transfer nodes for large/long-running transfers
- **Please use discretion**
  - *All* users get frustrated by sluggish interactive response

Login Node Guidelines

- Use *no more* than 50% of available cores
- Use *no more* than 25% of available memory
- Limit use of parallel "make"

```
% make -j 4 all
```

- NERSC will kill user processes if login nodes become unacceptably slow or unresponsive
- Terminate idle sessions of licensed software
  - IDL
  - MATLAB
  - Mathematica

Shell Initialization Files

- **Standard dot files are maintained by NERSC**
  - `.bashrc`, `.profile`, `.cshrc`, `.login`, etc.
  - Symbolic links to read-only files
- **Personal dot files**
  - Aliases, environment variables, modules, etc.
  - Use .ext suffix (".ext files") `.bashrc.ext`, etc.
- **Broken? Use "`fixdots`" to start over**
  - Creates `$HOME/KeepDots.<timestamp>`
  - Restores all dot files to default state
  - If `PATH` corrupted:
    `/usr/common/usg/bin/fixdots`
- **Use NIM to change default login shell**

NERSC Supported Software

- **NERSC provides a wide range of software**
  - Scientific Applications
    - VASP, Amber, NAMD, ABySS, ...
  - Compilers
    - Intel, GCC, PGI, Cray
  - Scripting Languages
    - perl, python, R - including common packages for each
  - Software Libraries (some maintained by Cray)
    - blas/lapack (MKL), boost, hdf5, netcdf, …
  - Development utilities
    - git, mercurial, cmake, …
  - Debuggers and Profilers
    - CrayPat, DDT, TotalView, gdb, MAP, darshan, IPM, VTune
  - Visualization
    - Visit, ParaView, VMD, ...
- **See complete list**

  http://www.nersc.gov/users/software/

Software is Managed by Modules

- **Identify the software you need**
  - Use the NERSC website

    http://www.nersc.gov/users/software/
  - Use `module avail`
    - *Lots* of output
      - All module output goes to stderr, not stdout
    - Each system has different modules!
- **Load the module**

```
% which idl
idl: Command not found.
% module load idl
% which idl
/usr/common/usg/idl/idl82/bin/idl
```

Loading Modules

- **Different module for each version of software**
  - Syntax: <name>/<version>
  - Default provided if no <version> supplied

```
% module avail idl
idl/7.1    idl/8.0    idl/8.2(default)
% module load idl/7.1
```

- **Load modules in every batch script**
  - Ensure correct run-time environment
  - Self-documenting for troubleshooting and reproducibility

Other Useful Module Commands

`module unload <modulename>`

- Remove the module from your environment

`module swap <module1> <module2>`

- Unload one module and replace it with another

```
module swap pgi gcc
```

`module list`

- See what modules you have loaded right now

`module show <modulename>`

- See what the module actually does

`module help <modulename>`

- Get more information about the software

Default Modules

- **When you login, many *default* modules are loaded automatically**
  - Usually foundational modules which are required to get proper function from the system
    - Build environment, required libraries and applications, batch environment
  - Use caution in unloading these
- **Swapping to functionally equivalent module may be OK**

```
hopper% module swap PrgEnv-pgi PrgEnv-gnu
```

- **Each NERSC system has different default modules**

Types of Modules

- **Applications**
  - VASP, amber, blast, …
  - Usually only set `PATH`, `LD_LIBRARY_PATH`
- **Libraries**
  - Set `LD_LIBRARY_PATH`
    - but probably not on Crays
  - Set "helper" environment variable for building software
    - Header/include file search paths
    - Library search paths
    - Library names

```
% module load hdf5
% mpicc mycode.f $HDF5
```

Cray Programming Environment

- **Compiler specific**

  PrgEnv-pgi, PrgEnv-intel, PrgEnv-cray, PrgEnv-gnu

  – Intel is default on Edison, PGI is default on Hopper

- ***Meta-modules***

  – Organize a set of modules

    - Compiler (intel, pgi, cray, gnu)
    - Libraries (including MPI) tuned for compiler

- **Swapping Programming Environments**

  `module swap PrgEnv-pgi PrgEnv-intel`

  – **swaps compiler**

  – ***no need to swap libraries!***

Compiler Wrappers

- **On Hopper / Edison:**
  - Defined by `PrgEnv-*` modules
  - `ftn` (fortran), `cc` (C), `CC` (C++)
  - Provides include header and library search paths for MPI, common math libraries (e.g., Cray libsci), Cray system software
  - Provides consistent level of optimization across compilers
- **Seldom need native compilers!**

CHOS Environment

- **Provides different OS environments**
  - Often different third-party software
    - Some software packages have specific OS requirements
      - Possibly due to validation requirements
- **Used on Genepool, PDSF, and Carver (retires in September 2015)**
- **Transparent**
  - Default configuration for most users
  - Alternate configurations for some users
- **Details on website**

http://www.nersc.gov/users/computational-systems/carver/user-environment/
http://www.nersc.gov/users/computational-systems/pdsf/software-and-tools/chos/

Resources

http://www.nersc.gov/users/software/nersc-user-environment/

http://www.nersc.gov/users/software/nersc-user-environment/modules/

http://www.nersc.gov/users/computational-systems/edison/programming

/http://www.nersc.gov/users/computational-systems/hopper/programming/

http://www.nersc.gov/users/computational-systems/carver/programming/

Thank you.